



# Tabela Hash

Prof. Jose J. Camata

Prof. Marcelo Caniato

Prof. Barbara Quintela

[camata@ice.ufjf.br](mailto:camata@ice.ufjf.br)

[marcelo.caniato@ice.ufjf.br](mailto:marcelo.caniato@ice.ufjf.br)

[barbara@ice.ufjf.br](mailto:barbara@ice.ufjf.br)

# Conteúdo

1. Definição
2. Funções Hash
3. Tratamento de Colisões
  - a. Encadeamento
  - b. Endereçamento Aberto

## Pesquisa em Memória Primária

- Estudo de como recuperar informação a partir de uma grande massa de informação previamente armazenada.
- A informação é dividida em registros.
- Cada registro possui uma chave para ser usada na pesquisa.
- Objetivo da pesquisa:
  - Encontrar uma ou mais ocorrências de registros com chaves iguais à chave de pesquisa.



## Escolha do Método de Pesquisa

- **Depende principalmente:**
  - Quantidade dos dados envolvidos.
  - Arquivo estar sujeito a inserções e retiradas frequentes.

*Se conteúdo do arquivo é estável, torna-se importante minimizar o tempo de pesquisa, sem preocupação com o tempo necessário para estruturar o arquivo*

# Algoritmos de Pesquisa

- É importante considerar os algoritmos de pesquisa como tipos abstratos de dados, com um conjunto de operações associado a uma estrutura de dados, de tal forma que haja uma independência de implementação para as operações.
- **Operações mais comuns:**
  - **Inicializar** a estrutura de dados.
  - **Pesquisar** um ou mais registros com determinada chave.
  - **Inserir** um novo registro.
  - **Retirar** um registro específico.
  - **Ordenar** um arquivo para obter todos os registros em ordem de acordo com a chave.
  - **Ajuntar** dois arquivos para formar um arquivo maior.

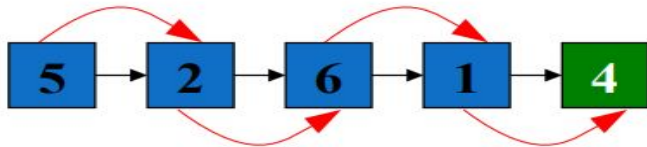
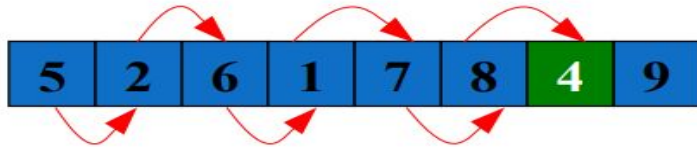


## Dicionário

- Nome comumente utilizado para descrever uma estrutura de dados para pesquisa.
- Dicionário é um tipo abstrato de dados com as operações:
  - *Inicializa*
  - *Insere*
  - *Pesquise*
  - *Remova*

# Pesquisa Sequencial

- Método de pesquisa mais simples:
  - *A partir do primeiro registro do dicionário, pesquise sequencialmente até encontrar a chave procurada; então pare.*
- Exemplo: procurar pela chave 4 (vetor e lista encadeada)



```
PesqSeq(A: vetor, n: int, chave: int)
  Para i ← 1 até n faça
    Se (A[i] == chave ) retorne i;
  Fim-Para
  retorne -1
```

## Pesquisa Sequencial

- Pesquisa retorna o índice do registro que contém a chave  $x$ ;
- Caso não esteja presente, o valor retornado é -1.
- A implementação não suporta mais de um registro com uma mesma chave.
- Para aplicações com esta característica é necessário incluir um argumento  $a$  mais na função Pesquisa para conter o índice a partir do qual se quer pesquisar.



## Pesquisa Sequencial: Análise

➤ Pesquisa com sucesso:

melhor caso :  $C(n) = 1$

pior caso :  $C(n) = n$

caso médio :  $C(n) = (n + 1)/2$

➤ O algoritmo de pesquisa sequencial é a melhor escolha para o problema de pesquisa em tabelas com até 25 registros.

## Outra Abordagem....

- Ao invés de organizar a tabela segundo o valor relativo de cada chave em relação aos demais, usa-se em conta somente o seu valor absoluto, interpretado como um valor numérico.

## Outra Abordagem....

➤ Exemplo:

- Uma empresa possui 100 funcionários. Cada funcionário é representado pela seguinte estrutura de dados:
- Os funcionários foram inseridos em tabela na ordem de contratação, onde o funcionário mais antigo tem matrícula 1 e o mais recente tem matrícula 100.

```
typedef struct {  
    int matricula;  
    string nome;  
    // Outros  
    atributos  
} Funcionario;
```

## Outra Abordagem....

### ➤ Exemplo (continuação):

- A pesquisa sobre um determinado funcionário pode ser feita de forma direta através de sua matrícula (chave de pesquisa)
- Pesquisa em T funcionário com matrícula 3:
  - $F = T[3]$

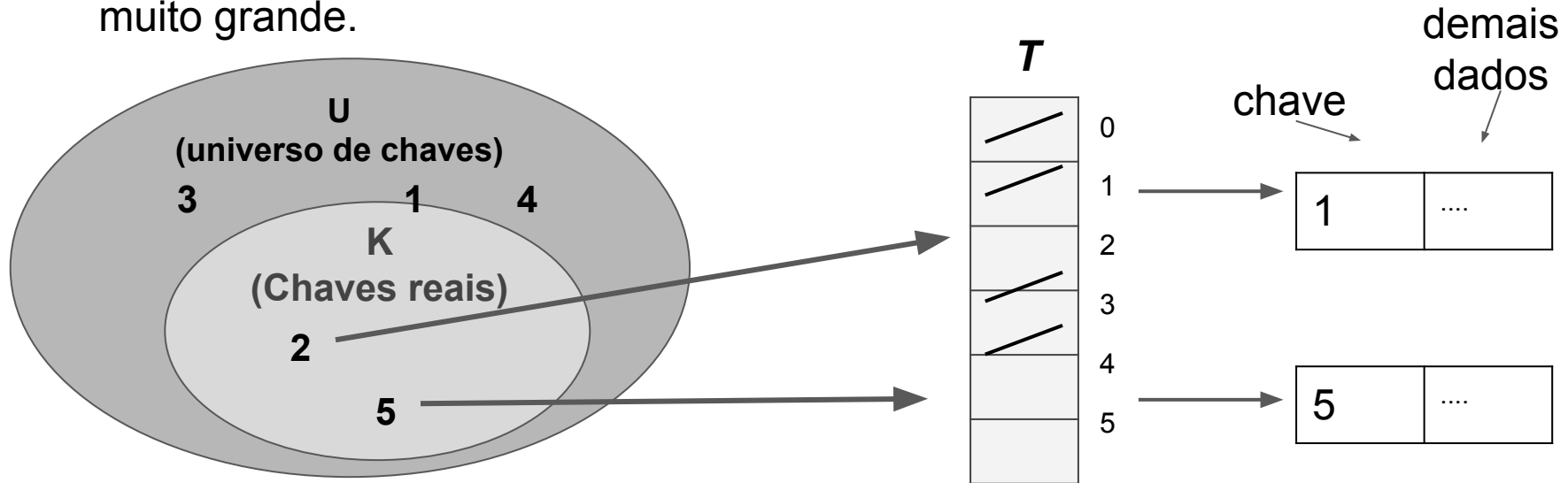
### ➤ Note que a pesquisa ocorre com endereçamento direto!

- Custo da pesquisa:  $O(1)$

T	
1	Matrícula: 1 Nome: Jose
2	Matrícula: 2 Nome: Matria
3	Matrícula: 3 Nome: João
.	.
.	.
.	.
100	Matrícula: 100 Nome: Fulano

# Tabelas de Endereçamento Direto

- Técnica simples que funciona bem quando o universo  $U$  de chaves é razoavelmente pequeno.
- Na Tabela de endereçamento diretos  $T=[0,...,m-1]$ , cada posição corresponde a uma chave no universo  $U = \{0, 1, ..., m-1\}$ , onde  $m$  não é muito grande.



## Tabelas de Endereçamento Direto

- O ponto negativo do endereçamento direto é óbvio:
  - Se o universo  $U$  é grande, armazenar uma tabela  $T$  de tamanho  $|U|$  pode ser impraticável ou mesmo impossível!!
- Exemplo:
  - Imagine agora se, ao invés de usar a matrícula dos funcionários como chave, fosse usado o seu CPF.
  - As chaves possíveis do universo  $U$  iriam variar de 000.000.000 até 999.999.999 (excluído os dois dígitos verificadores).
  - Assim,  $T$  teria que ser alocado com 1 Bi de posições mas somente 100 seriam utilizadas.

## Tabela de Espalhamento (Hash)

- Os registros armazenados em uma tabela são diretamente endereçados a partir de uma transformação aritmética sobre a chave de pesquisa.
- **Acesso direto**, mas **endereçamento indireto**.
  - **Função de dispersão** (função hash)  $h(k) \neq k$ , em geral
  - Resolve uso ineficiente de espaço de armazenamento

## Tabelas de Espalhamento (Hash)

- Com endereçamento direto, um elemento com a chave  $k$  é armazenado na posição  $k$ .
- Com hash, esse elemento é armazenado na posição  $h(k)$ ; isto é, usamos uma função *hash*  $h$  para calcular a posição da chave  $k$  na tabela  $T$ .
- Qualquer que seja a função hash, algumas colisões poderão fatalmente ocorrer, e tais colisões têm de ser resolvidas de alguma forma.
- Mesmo que se obtenha uma função de transformação que distribua os registros de forma uniforme entre as entradas da tabela, existe uma alta probabilidade de haver colisões.



## Tabelas de Espalhamento (Hash)

- O paradoxo do aniversário (Feller, 1968, p. 33), diz que em um grupo de 23 ou mais pessoas, juntas ao acaso, existe uma chance maior do que 50% de que 2 pessoas comemorem aniversário no mesmo dia.
- Assim, se for utilizada uma função de transformação uniforme que enderece 23 chaves randômicas em uma tabela de tamanho 365, a probabilidade de que haja colisões é maior do que 50%.
- A probabilidade  $p$  de se inserir  $N$  itens consecutivos sem colisão em uma tabela de tamanho  $M$  é:

$$p = \frac{M-1}{M} \times \frac{M-2}{M} \times \dots \times \frac{M-N+1}{M} = \prod_{i=1}^N \frac{M-i+1}{M} = \frac{M!}{(M-N)!M^N}$$

## Tabelas de Espalhamento (Hash)

Alguns valores de  $p$  para diferentes valores de  $N$ , onde  $M = 365$ .

$N$	$p$
10	0,883
22	0,524
23	0,493
30	0,303

Para  $N$  pequeno a probabilidade  $p$  pode ser aproximada por  $p \approx \frac{N(N-1)}{730}$ . Por exemplo, para  $N = 10$  então  $p \approx 87,7\%$ .

# Funções Hash

## Funções hash

- Uma função *hash* deve mapear chaves em inteiros dentro do intervalo  $[0 \dots M - 1]$ , onde  $M$  é o tamanho da tabela.
- **A função de transformação ideal é aquela que:**
  - Seja simples de ser computada.
  - Para cada chave de entrada, qualquer uma das saídas possíveis é **igualmente provável** de ocorrer.

# Funções hash

## ➤ **Hashing perfeita:**

- Transforma diferentes chaves em diferentes posições **sem** colisões
- Tabela tem que conter mesmo número de posições que o número de elementos que sofreram *hashing*
- **Problema:**
  - Pode-se não conhecer a priori o número total de elementos
  - O número possível de elementos pode ser muito maior do que o número total de elementos

# Funções Hash

- Métodos:
  - Divisão
  - Multiplicação
  - Compressão de Chaves Alfanuméricas
  - Enlaçamento
    - Deslocado
    - Limite
  - Meio-Quadrado
  - Extração
  - Transformação de Raiz

## Método da divisão

- Mapeia uma chave inteira  $k$  para uma de  $m$  posições, tomando o resto da divisão de  $k$  por  $m$ :

$$h(k) = k \bmod m$$

- Cuidado na escolha do valor de  $m$ !!!
  - $m$  não deve ser potência de 2.
  - $m$  como um número primo não muito próximo de uma potência de 2 é uma boa escolha.
- Se  $m$  não é um número primo pode-se utilizar função:
  - $h(k) = (k \bmod p) \bmod m$ , onde  $p$  é um número primo maior que  $m$ .

## Método da multiplicação

- Realizado em duas etapas:
  - Multiplicamos a chave  $k$  por uma constante  $A$  na faixa  $0 < A < 1$  e extraímos a parte fracionária.
  - Multiplicamos esse valor por  $m$  e tomamos o piso do resultado.

$$h(k) = \text{floor}(m(kA \bmod 1))$$

onde  $kA \bmod 1$  significa a parte fracionária de  $kA$ .

- A escolha de  $m$  não é crítico. Pode ser potência de 2.
- O método depende da escolha de  $A$ .
  - Escolha ótima depende dos dados.
  - Knuth [1971] sugere:  $A = (\text{sqrt}(5) - 1) / 2$



# Enlaçamento

- Neste método a chave é dividida em diversas partes que são combinadas ou “enlaçadas” e transformadas para criar o endereço.
- Existem dois tipos de enlaçamento:
  - Enlaçamento deslocado
  - Enlaçamento limite

## Enlaçamento Deslocado

1. Chave é dividida em partes homogêneas
2. Uma parte é colocada embaixo da outra
3. Enlaçamento deslocado soma estas partes.
4. Aplica-se o método de divisão no resultado

**Exemplo:** Código pessoal 123-456-789

123

456

789

1. Dividir em partes e colocar uma embaixo da outra:
2. Somar as partes:  $123+456+789 = 1368$
3. Aplicar o método da divisão:
  - a. Supondo  $m = 1000$
  - b.  $1368 \bmod 1000 = 368$

## Enlaçamento Limite

- As partes da chave são colocadas em ordem inversa.
- Considerando as mesmas divisões do código 123-456-789.
- Alinha-se as partes sempre invertendo as divisões da seguinte forma 321-654-987.
- O resultado da soma é 1566.
- Esse valor pode usar o método da divisão

## Meio-Quadrado

- A chave é elevada ao quadrado e a parte do resultado é usada como endereço.
- Exemplo:  $k = 3121$ 
  - $3121^2 = 97\textcolor{red}{406}41$
  - Em uma tabela com  $m=1000$ , pode-se utilizar os 3 dígitos do meio:
    - $h(3121) = 406$
  - Uma máscara binária pode ser usada para obter a posição:
    - Se tamanho da tabela é 1024 (em binário 10000000000)
    - $3121^2$  em binário é igual a 1001010 $\textcolor{red}{0101000010}$ 11000001
    - $\textcolor{red}{0101000010}$ , que é igual a 322, pode ser extraído usando máscara e uma operação de deslocamento

## Extração

- Neste método somente uma parte da chave é usada para criar o endereço
- Para o código 123-456-789 pode-se usar os primeiros ou os últimos 4 dígitos ou outro tipo de combinação como 1289.
- Somente uma porção da chave é usada.

## Transformação de raiz

- A chave é transformada para outra base numérica.
- O valor obtido é aplicado no método da divisão para obter o endereço.
- Exemplo:
  - $k = 345$  na base decimal.
  - Valor de  $k$  na base nona: 423
  - Aplica método de divisão:
    - Se  $m = 100$ ,  $h(345) = 423 \bmod 100 = 23$ .
  - **Não evita colisões!**

## Exercício:

1. Considere uma tabela hash de tamanho igual a 7. Usando o método da divisão, faça a inserção dos seguintes números: 1, 5, 10, 20, 25, 24.

T

	0
	1
	2
	3
	4
	5
	6

## Exercício:

1. Considere uma tabela hash de tamanho igual a 7. Usando o método da divisão, faça a inserção dos seguintes números: 1, 5, 10, 20, 25, 24.

- Inserindo 1 em T:

- $h(1) = 1 \% 7 = 1$
- $T[h(1)] = 1$



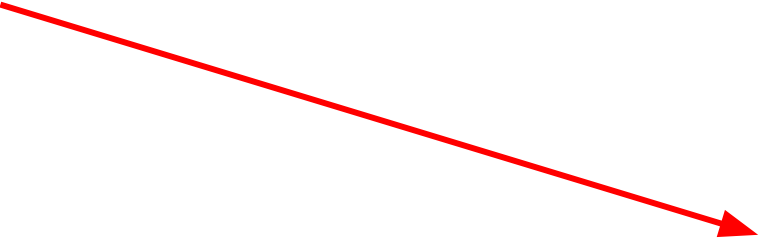
T	
	0
1	1
	2
	3
	4
	5
	6



## Exercício:

1. Considere uma tabela hash de tamanho igual a 7. Usando o método da divisão, faça a inserção dos seguintes números: 1, 5, 10, 20, 25, 24.

- Inserindo 5 em T:
  - $h(5) = 5 \% 7 = 5$
  - $T[h(5)] = 5$

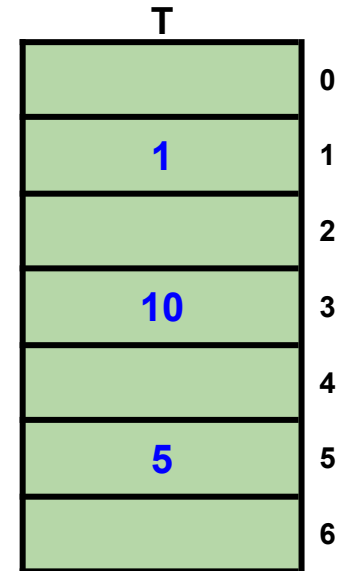


T	
	0
1	1
	2
	3
	4
5	5
	6

## Exercício:

1. Considere uma tabela hash de tamanho igual a 7. Usando o método da divisão, faça a inserção dos seguintes números: 1, 5, **10**, 20, 25, 24.

- Inserindo 10 em T:
  - $h(10) = 10 \% 7 = 3$
  - $T[h(10)] = 10$

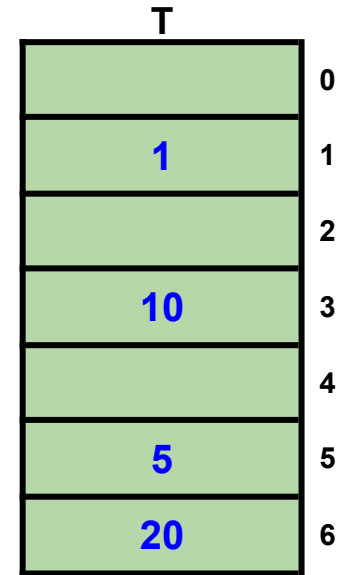


T	
	0
1	1
	2
10	3
	4
5	5
	6

## Exercício:

1. Considere uma tabela hash de tamanho igual a 7. Usando o método da divisão, faça a inserção dos seguintes números: 1, 5, 10, **20**, 25, 24.

- Inserindo 20 em T:
  - $h(20) = 20 \% 7 = 6$
  - $T[h(20)] = 20$

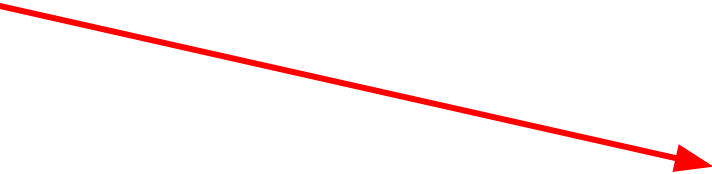


T	
	0
1	1
	2
10	3
	4
5	5
20	6

## Exercício:

1. Considere uma tabela hash de tamanho igual a 7. Usando o método da divisão, faça a inserção dos seguintes números: 1, 5, 10, 20, **25**, 24.

- Inserindo 25 em T:
  - $h(25) = 25 \% 7 = 4$
  - $T[h(25)] = 25$




T		
		0
1		1
		2
10		3
25		4
5		5
20		6

## Exercício:

1. Considere uma tabela hash de tamanho igual a 7. Usando o método da divisão, faça a inserção dos seguintes números: 1, 5, 10, 20, 25, **24**.

- Inserindo 24 em T:
  - $h(24) = 24 \% 7 = 3$
  - $T[h(24)] = 24$

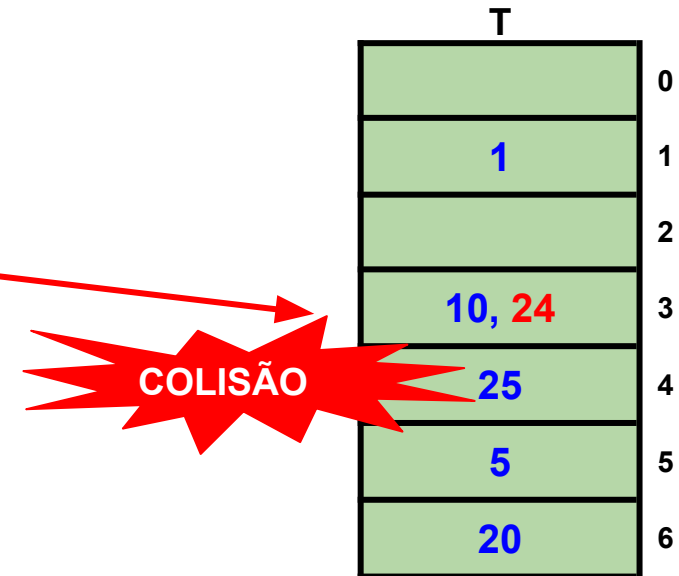


T	
	0
1	1
	2
10	3
25	4
5	5
20	6

## Exercício:

1. Considere uma tabela hash de tamanho igual a 7. Usando o método da divisão, faça a inserção dos seguintes números: 1, 5, 10, 20, 25, 24.

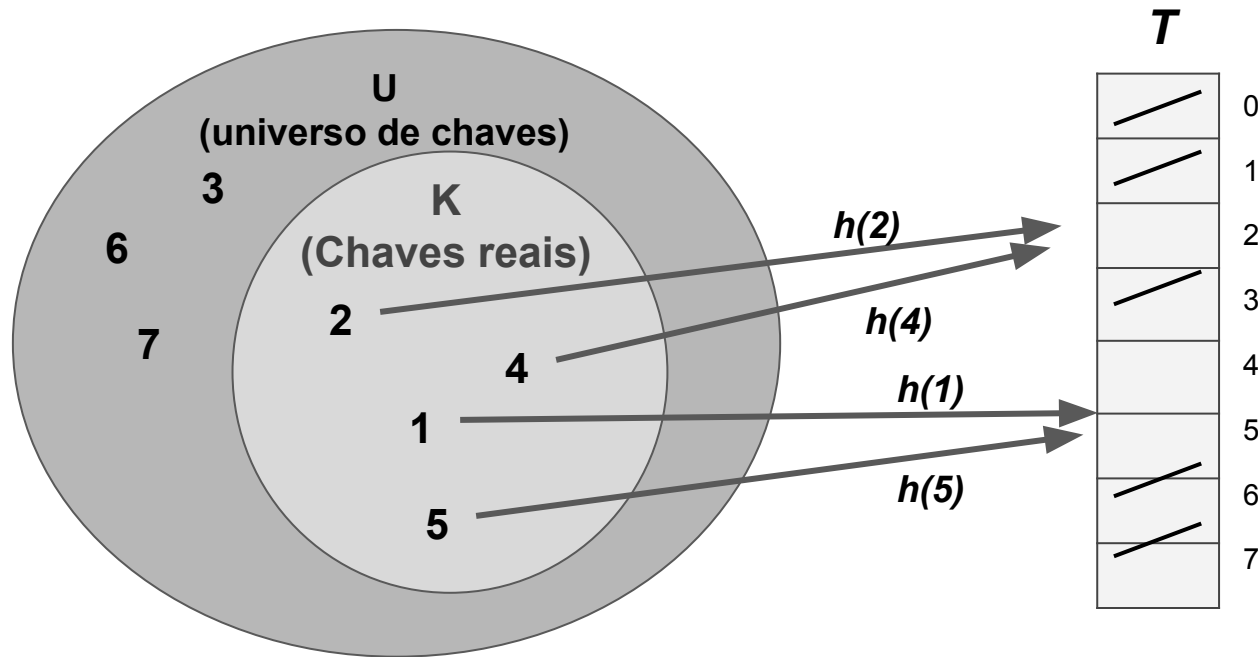
- Inserindo 24 em T:
  - $h(24) = 24 \% 7 = 3$
  - $T[h(24)] = 24$



# Tratamento de Colisões

# Tratamento de Colisões

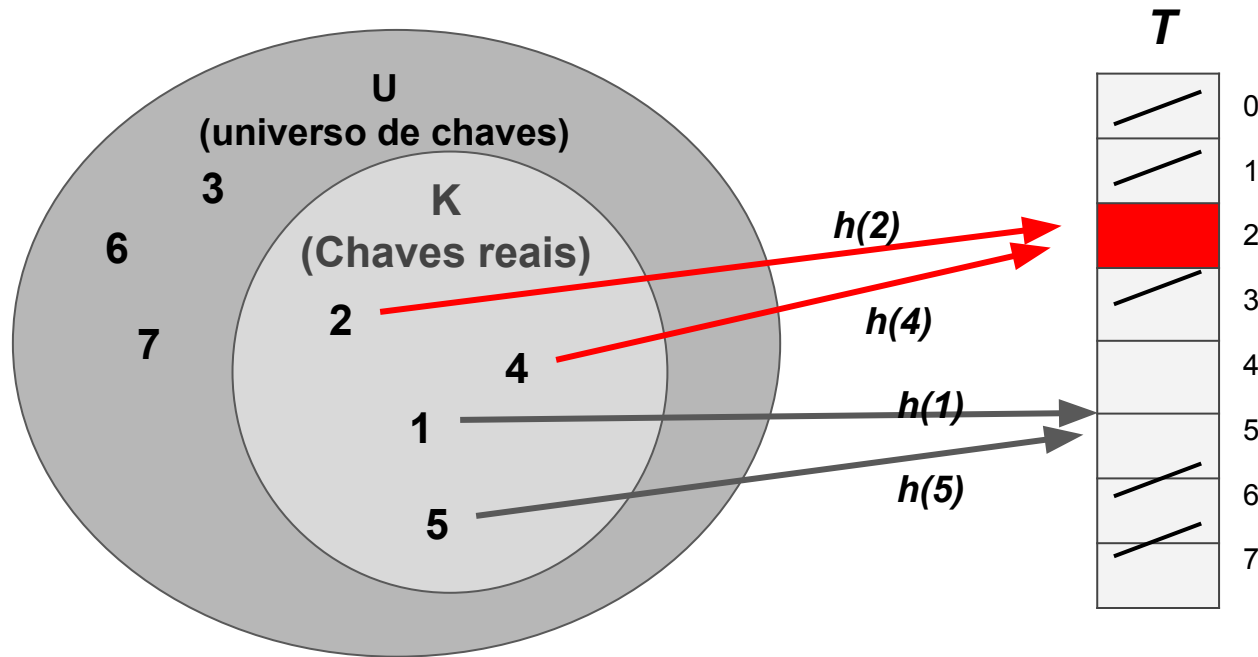
- O que fazer quando duas chaves são mapeadas para a mesma posição da tabela?





# Tratamento de Colisões

- O que fazer quando duas chaves são mapeadas para a mesma posição da tabela?



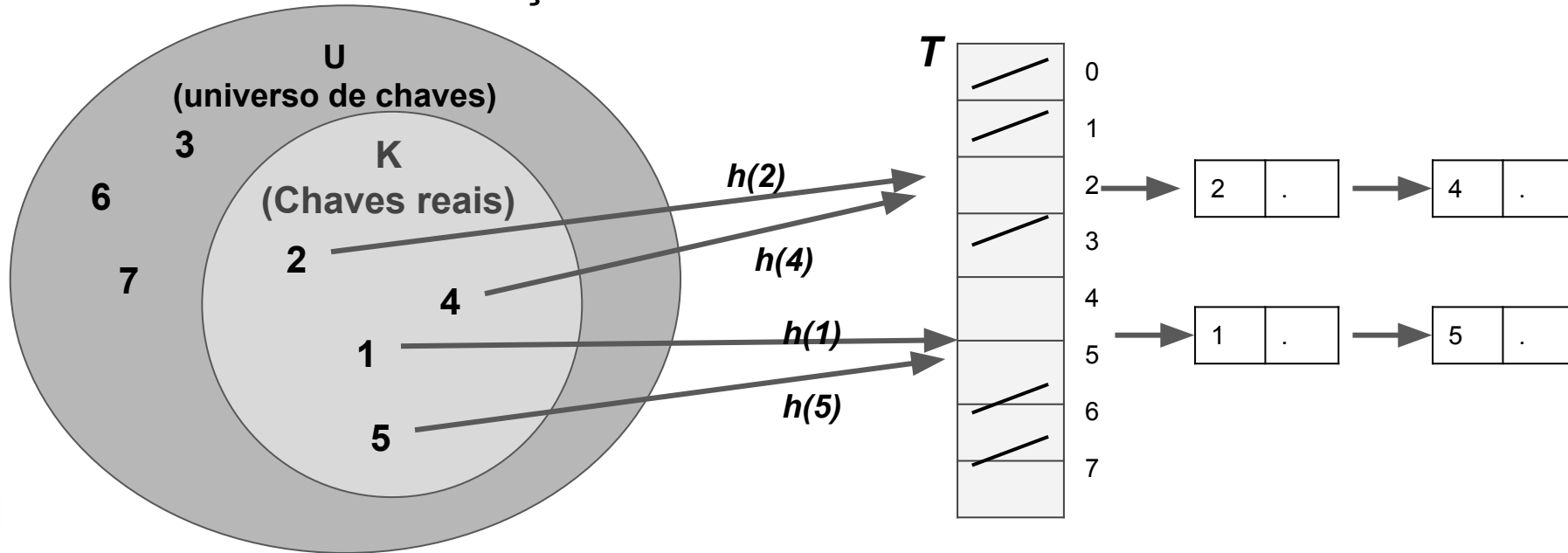
# Tratamento de Colisões

- Encadeamento
  - Separado
  - Coalescido
  - Coalescido com porção
- Endereçamento Aberto
  - Sondagem Linear
  - Sondagem Quadrática
  - Sondagem Dupla

## Encadeamento Separado

### Tratamento de Colisões por Lista Encadeada

- Uma das formas de resolver as colisões é construir uma lista linear encadeada para cada endereço da tabela. Assim, todas as chave  $k$  com mesmo endereço são encadeadas em uma lista linear.





## Encadeamento Externo

### Tratamento de Colisões por Lista Encadeada

- Manter  $m$  listas encadeadas, uma para cada possível endereço base.
- A tabela base não possui nenhum registro, apenas os ponteiros para as listas encadeadas.
- Por isso chamamos de encadeamento exterior: **a tabela base não armazena nenhum registro!!**

Cada nó da lista encadeada armazena:

- *Registro*
- *Ponteiro para o próximo nó da lista*

## Encadeamento Coalescido

- Utilizado quando não é desejável a manutenção de uma estrutura externa a uma tabela hash.
  - Quando não se pode permitir que o espaço de registros cresça indefinidamente.
- Cada posição na tabela possui espaço para a chave e um ponteiro que aponta para a posição da próxima chave.
- Quando ocorre uma nova colisão, a nova chave é inserida no primeiro espaço vazio a partir, por exemplo, do compartimento onde ocorreu a colisão, ou do final da tabela.

Cada posição da tabela armazena dois atributos:

- INFO: armazena informação da chave
- NEXT: índice para a próxima chave caso haja colisão.
  - -2 pode indicar posição livre.
  - -1 pode indicar final de uma cadeia.

INFO	NEXT

# Encadeamento Coalescido

➤ *Exemplo de Inserção:*

Tabela Vazia

*T*

0		-2
1		-2
2		-2
3		-2
4		-2
5		-2
6		-2
7		-2
8		-2
9		-2

# Encadeamento Coalescido

➤ *Exemplo de Inserção:*

1. Inserção de A2:

*T*

0		-2
1		-2
2	<b>A2</b>	-1
3		-2
4		-2
5		-2
6		-2
7		-2
8		-2
9		-2

# Encadeamento Coalescido

## ➤ *Exemplo de Inserção:*

1. Inserção de A2:
2. Inserção de A3:

***T***

0		-2
1		-2
2	<b>A2</b>	-1
3	<b>A3</b>	-1
4		-2
5		-2
6		-2
7		-2
8		-2
9		-2



# Encadeamento Coalescido

## ➤ *Exemplo de Inserção:*

1. Inserção de A2:
2. Inserção de A3:
3. Inserção de A5:

*T*

0		-2
1		-2
2	<b>A2</b>	-1
3	<b>A3</b>	-1
4		-2
5	<b>A5</b>	-1
6		-2
7		-2
8		-2
9		-2

# Encadeamento Coalescido

## ➤ *Exemplo de Inserção:*

1. Inserção de A2:
2. Inserção de A3:
3. Inserção de A5:
4. Inserção de B5:

*T*

0		-2
1		-2
2	<b>A2</b>	-1
3	<b>A3</b>	-1
4		-2
5	<b>A5</b>	-1
6		-2
7		-2
8		-2
9		-2

# Encadeamento Coalescido

## ➤ *Exemplo de Inserção:*

1. Inserção de A2:
2. Inserção de A3:
3. Inserção de A5:
4. Inserção de B5:
5. **Colisão com A5**

<i>T</i>	
0	-2
1	-2
2	A2 -1
3	A3 -1
4	-2
5	A5 -1
6	-2
7	-2
8	-2
9	-2

# Encadeamento Coalescido

## ➤ Exemplo de Inserção:

1. Inserção de A2:
2. Inserção de A3:
3. Inserção de A5:
4. Inserção de B5:
5. **Colisão com A5**
6. **Insere na primeira posição vazia a partir do final da tabela**

<i>T</i>	
0	-2
1	-2
2	A2 -1
3	A3 -1
4	-2
5	A5 9
6	-2
7	-2
8	-2
9	B5 -1




# Encadeamento Coalescido

## ➤ *Exemplo de Inserção:*

1. Inserção de A2:
2. Inserção de A3:
3. Inserção de A5:
4. Inserção de B5:
5. Colisão com A5
6. Insere na primeira posição vazia a partir do final da tabela
7. Inserir A9

*T*

0		-2
1		-2
2	<b>A2</b>	-1
3	<b>A3</b>	-1
4		-2
5	<b>A5</b>	<b>9</b>
6		-2
7		-2
8		-2
9	<b>B5</b>	-1




# Encadeamento Coalescido

## ➤ Exemplo de Inserção:

1. Inserção de A2:
2. Inserção de A3:
3. Inserção de A5:
4. Inserção de B5:
5. Colisão com A5
6. Insere na primeira posição vazia a partir do final da tabela
7. Inserir A9
8. **Colisão com B5**

*T*

0		-2
1		-2
2	A2	-1
3	A3	-1
4		-2
5	A5	9
6		-2
7		-2
8		-2
9	B5	-1



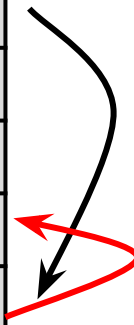
# Encadeamento Coalescido

## ➤ Exemplo de Inserção:

1. Inserção de A2:
2. Inserção de A3:
3. Inserção de A5:
4. Inserção de B5:
5. Colisão com A5
6. Insere na primeira posição vazia a partir do final da tabela, ie, em T[9]
7. Inserir A9
8. **Colisão com B5**
9. Insere na primeira posição vazia a partir do final da tabela, ie, em T[8]

**T**

0		-2
1		-2
2	<b>A2</b>	-1
3	<b>A3</b>	-1
4		-2
5	<b>A5</b>	<b>9</b>
6		-2
7		-2
8	<b>A9</b>	-1
9	<b>B5</b>	8

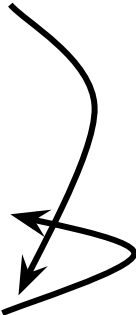


# Encadeamento Coalescido

## ➤ *Exemplo de Inserção:*

1. Inserção de A2
2. Inserção de A3
3. Inserção de A5
4. Inserção de B5
5. Colisão com A5
6. Insere na primeira posição vazia a partir do final da tabela, ie, em T[9]
7. Inserção de A9
8. Colisão com B5
9. Insere na primeira posição vazia a partir do final da tabela, ie, em T[8]
10. Inserção de B2

<i>T</i>	
0	-2
1	-2
2	<b>A2</b> -1
3	<b>A3</b> -1
4	-2
5	<b>A5</b> 9
6	-2
7	-2
8	<b>A9</b> -1
9	<b>B5</b> 8





# Encadeamento Coalescido

## ➤ Exemplo de Inserção:

1. Inserção de A2
2. Inserção de A3
3. Inserção de A5
4. Inserção de B5
5. Colisão com A5
6. Insere na primeira posição vazia a partir do final da tabela, ie, em  $T[9]$
7. Inserção de A9
8. Colisão com B5
9. Insere na primeira posição vazia a partir do final da tabela, ie, em  $T[8]$
10. Inserção de B2
11. **Colisão com A2**

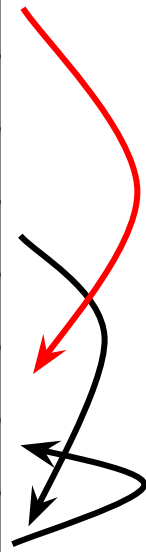
$T$	
0	-2
1	-2
2	<b>A2</b> -1
3	<b>A3</b> -1
4	-2
5	<b>A5</b> 9
6	-2
7	-2
8	<b>A9</b> -1
9	<b>B5</b> 8

# Encadeamento Coalescido

## ➤ Exemplo de Inserção:

1. Inserção de A2
2. Inserção de A3
3. Inserção de A5
4. Inserção de B5
5. Colisão com A5
6. Insere na primeira posição vazia a partir do final da tabela, ie, em  $T[9]$
7. Inserção de A9
8. Colisão com B5
9. Insere na primeira posição vazia a partir do final da tabela, ie, em  $T[8]$
10. Inserção de B2
11. **Colisão com A2**
12. **Insere na primeira posição vazia a partir do final da tabela, ie, em  $T[7]$**

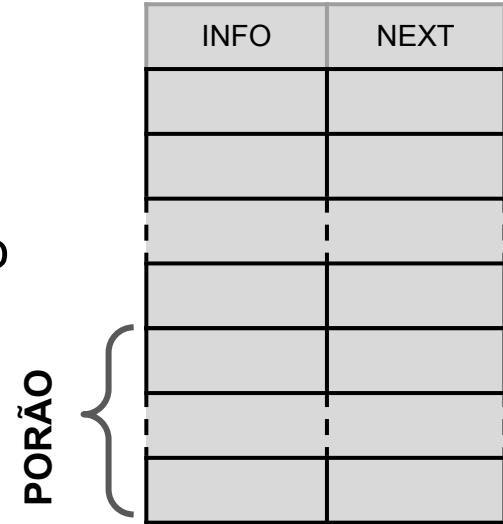
$T$	
0	-2
1	-2
2	<b>A2</b> 7
3	<b>A3</b> -1
4	-2
5	<b>A5</b> 9
6	-2
7	<b>B2</b> -1
8	<b>A9</b> -1
9	<b>B5</b> 8



The diagram illustrates the insertion process. A red arrow points from the entry 'B2' at index 7 to the entry 'A2' at index 2, indicating a collision. A black arrow points from the entry 'A9' at index 8 to the entry 'B5' at index 9, indicating a collision. Another black arrow points from the entry 'B5' at index 9 to the entry 'A2' at index 2, indicating a collision. A final black arrow points from the entry 'A2' at index 2 to the entry 'B2' at index 7, indicating a collision.

# Encadeamento Coalescido com porão

- No encadeamento coalescido, o tamanho da tabela limita o número de chaves que podem ser escrutinadas na tabela!
- Outra opção de solução é separar uma área de transbordamento, conhecida como porão, para o tratamento das colisões
- Pode ser alocada dinamicamente



# Coalescido com Porão

➤ *Exemplo de Inserção:*

- *A5,A2,A3,B5,A9,B2,B9,C2*

Tabela Vazia

*T*

0		-2
1		-2
2		-2
3		-2
4		-2
5		-2
6		-2
7		-2
8		-2
9		-2
10		-2
11		-2
12		-2

# Coalescido com Porão

➤ *Exemplo de Inserção:*

- **A5, A2, A3, B5, A9, B2, B9, C2**

1. Inserção de A5

***T***

0		-2
1		-2
2		-2
3		-2
4		-2
5	<b>A5</b>	-1
6		-2
7		-2
8		-2
9		-2
10		-2
11		-2
12		-2

# Coalescido com Porão

➤ *Exemplo de Inserção:*

○ *A5, **A2**, A3, B5, A9, B2, B9, C2*

1. Inserção de A5
2. Inserção de A2

*T*

0		-2
1		-2
2	A2	-1
3		-2
4		-2
5	A5	-1
6		-2
7		-2
8		-2
9		-2
10		-2
11		-2
12		-2

# Coalescido com Porão

➤ *Exemplo de Inserção:*

○ *A5, A2, **A3**, B5, A9, B2, B9, C2*

1. Inserção de A5
2. Inserção de A2
3. Inserção de A3

	<i>T</i>	
0		-2
1		-2
2	A2	-1
3	<b>A3</b>	<b>-2</b>
4		-2
5	A5	-1
6		-2
7		-2
8		-2
9		-2
10		-2
11		-2
12		-2

# Coalescido com Porão

➤ *Exemplo de Inserção:*

○ *A5, A2, A3, **B5**, A9, B2, B9, C2*

1. Inserção de A5
2. Inserção de A2
3. Inserção de A3
4. Inserção de B5

	<i>T</i>	
0		-2
1		-2
2	A2	-1
3	A3	-2
4		-2
5	A5	-1
6		-2
7		-2
8		-2
9		-2
10		-2
11		-2
12		-2



# Coalescido com Porão

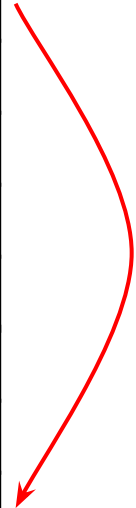
➤ *Exemplo de Inserção:*

○ *A5, A2, A3, **B5**, A9, B2, B9, C2*

1. Inserção de A5
2. Inserção de A2
3. Inserção de A3
4. Inserção de B5
5. Colisão com A5:
  - a. Insere na primeira posição livre do porão.

***T***

0		-2
1		-2
2	A2	-1
3	A3	-2
4		-2
5	A5	12
6		-2
7		-2
8		-2
9		-2
10		-2
11		-2
12	B5	-1



# Coalescido com Porão

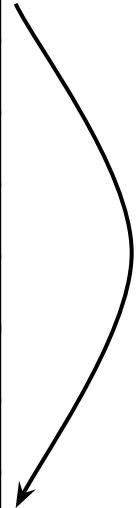
➤ *Exemplo de Inserção:*

- *A5, A2, A3, B5, **A9**, B2, B9, C2*

1. Inserção de A5
2. Inserção de A2
3. Inserção de A3
4. Inserção de B5
5. Colisão com A5:
  - a. Insere na primeira posição livre do porão.
6. Inserção de A9

*T*

0		-2
1		-2
2	A2	-1
3	A3	-2
4		-2
5	A5	12
6		-2
7		-2
8		-2
9	A9	-1
10		-2
11		-2
12	B5	-1



# Coalescido com Porão

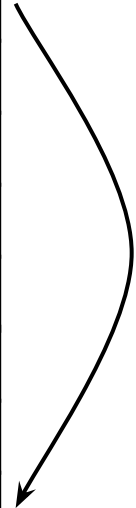
## ➤ Exemplo de Inserção:

- A5, A2, A3, B5, A9, **B2**, B9, C2

1. Inserção de A5
2. Inserção de A2
3. Inserção de A3
4. Inserção de B5
5. Colisão com A5:
  - a. Insere na primeira posição livre do porão.
6. Inserção de A9
7. Inserção de B2

*T*

0		-2
1		-2
2	A2	-1
3	A3	-2
4		-2
5	A5	12
6		-2
7		-2
8		-2
9	A9	-1
10		-2
11		-2
12	B5	-1



# Coalescido com Porão

## ➤ Exemplo de Inserção:

- A5, A2, A3, B5, A9, **B2**, B9, C2

1. Inserção de A5
2. Inserção de A2
3. Inserção de A3
4. Inserção de B5
5. Colisão com A5:
  - a. Insere na primeira posição livre do porão.
6. Inserção de A9
7. Inserção de B2
8. Colisão com A2
  - a. Insere na primeira posição livre do porão.

*T*

0		-2
1		-2
2	A2	11
3	A3	-2
4		-2
5	A5	12
6		-2
7		-2
8		-2
9	A9	-1
10		-2
11	B2	-1
12	B5	-1

# Coalescido com Porão

## ➤ Exemplo de Inserção:

- A5,A2,A3,B5,A9,B2,**B9**,C2

1. Inserção de A5
2. Inserção de A2
3. Inserção de A3
4. Inserção de B5
5. Colisão com A5:
  - a. Insere na primeira posição livre do porão.
6. Inserção de A9
7. Inserção de B2
8. Colisão com A2
  - a. Insere na primeira posição livre do porão.
9. Inserção de B9

*T*

0		-2
1		-2
2	A2	11
3	A3	-2
4		-2
5	A5	12
6		-2
7		-2
8		-2
9	A9	-1
10		-2
11	B2	-1
12	B5	-1

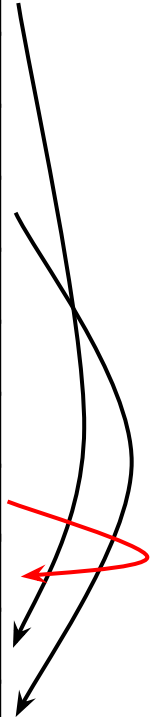
# Coalescido com Porão

## ➤ Exemplo de Inserção:

○ A5, A2, A3, B5, A9, B2, **B9**, C2

1. Inserção de A5
2. Inserção de A2
3. Inserção de A3
4. Inserção de B5
5. Colisão com A5:
  - a. *Insere na primeira posição livre do porão.*
6. Inserção de A9
7. Inserção de B2
8. Colisão com A2
  - a. *Insere na primeira posição livre do porão.*
9. Inserção de B9
10. Colisão com A9
  - a. *Insere na primeira posição livre do porão*

	<i>T</i>	
0		-2
1		-2
2	A2	11
3	A3	-2
4		-2
5	A5	12
6		-2
7		-2
8		-2
9	A9	10
10	<b>B9</b>	-1
11	B2	-1
12	B5	-1



# Coalescido com Porão

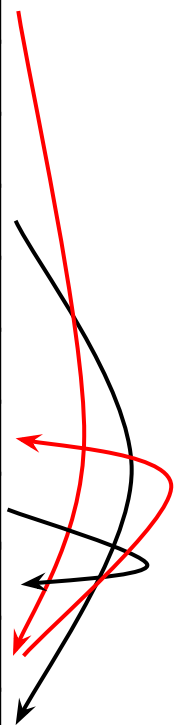
## ➤ Exemplo de Inserção:

○ **A5,A2,A3,B5,A9,B2,B9,C2**

1. Inserção de A5
2. Inserção de A2
3. Inserção de A3
4. Inserção de B5
5. Colisão com A5:
  - a. *Insera na primeira posição livre do porão.*
6. Inserção de A9
7. Inserção de B2
8. Colisão com A2:
  - a. *Insera na primeira posição livre do porão.*
9. Inserção de B9
10. Colisão com A9:
  - a. *Insera na primeira posição livre do porão*
11. *Inserção de C2*

**T**

0		-2
1		-2
2	A2	11
3	A3	-2
4		-2
5	A5	12
6		-2
7		-2
8		-2
9	A9	10
10	B9	-1
11	B2	-1
12	B5	-1





# Remoção por Encadeamento

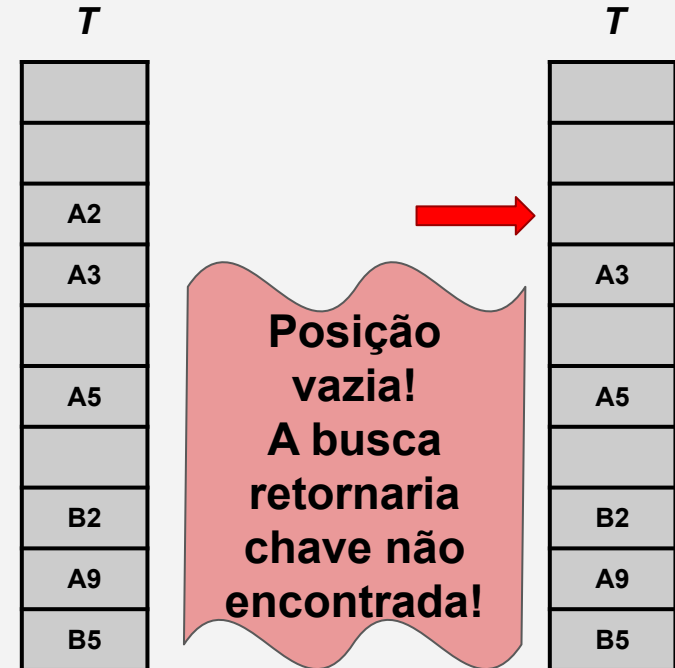
- Encadeamento por Lista Encadeada:
  - Remover a chave  $k$  na lista encadeada  $T[h(k)]$ .
- Encadeamento Coalescido/Coalescido com porção:
  - Remoção exige mais cuidados.
  - A remoção de um elemento da lista encadeada armazenada em  $T$  pode gerar resultados errôneos na busca por uma chave

## Possível Solução:

Manter chaves removidas, marcando as posições como elementos REMOVIDOS.

*Exemplo:*

*Remoção de A2  
Pesquisa por B2*



## Encadeamento

- A eficiência desta implementação depende do tamanho  $M$  da tabela e do número  $N$  de chaves existentes na tabela.
- O valor  $\alpha = N/M$  é o **fator de carga** da tabela.
- O fator  $\alpha$  é também o ***tamanho médio*** da lista encadeada.
  - Custo médio de busca  $\approx 1 + \alpha/2$ .
  - Ideal  $\alpha$  ser constante.

# Tratamento de Colisões

- Encadeamento
  - Separado
  - Coalescido
  - Coalescido com porão
- **Endereçamento Aberto**
  - Sondagem Linear
  - Sondagem Quadrática
  - Sondagem Dupla

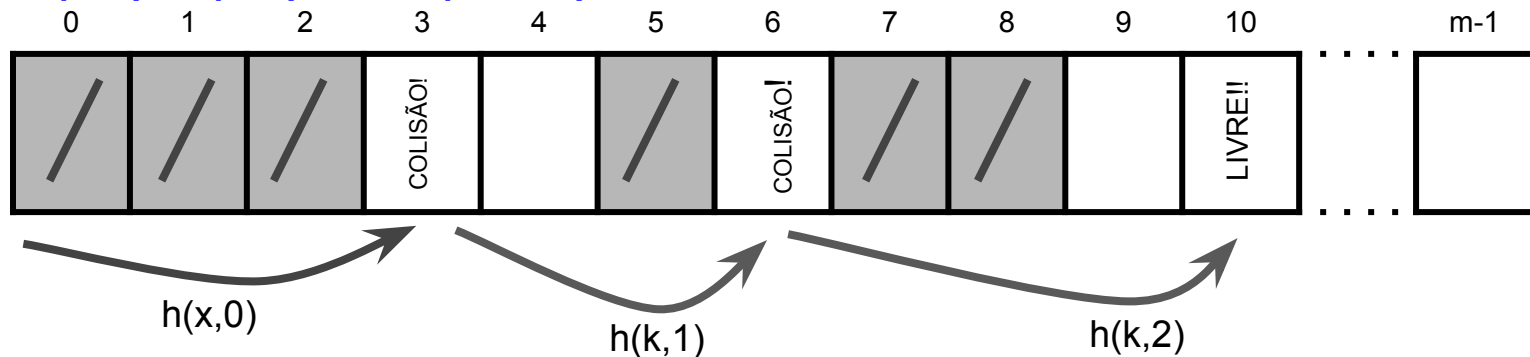
# Endereçamento Aberto

- **Motivação:** as abordagens anteriores utilizam ponteiros/índices nos encadeamentos
  - Aumento no consumo de espaço
- **Ideia básica** do Endereçamento Aberto:
  - Armazenar as chaves sinônimas sem qualquer informação adicional
  - Quando houver colisão, determina-se por cálculo o novo endereço pela qual a próxima posição deva ser examinada.
  - Busca com sucesso quando um compartimento é encontrado com a chave procurada
  - Busca sem sucesso seria a computação de um um compartimento vazio ou a exaustão da tabela

Endereçamento Aberto:

## Funcionamento

- Quando uma chave  $k$  é endereçada para uma entrada da tabela que já esteja ocupada (colisão), uma sequência de localizações alternativas  $h(k,1)$ ,  $h(k,2)$ , ...,  $h(k,m-1)$  é escolhida.



- Para cada chave  $k$ , a função  $h$  deve ser capaz de fornecer uma permutação de endereços base no conjunto  $\{0, \dots, m-1\}$ .

## Endereçamento Aberto: Funções Hash

- Existem várias propostas para a escolha de localizações alternativas:
  - Sondagem Linear
  - Sondagem Quadrática
  - Sondagem Dupla

Endereçamento Aberto:

## Sondagem Linear

- Dada uma função hash comum  $h'$ , o método de sondagem linear usa a função:

$$h(x,i) = (h'(k) + i) \bmod m \quad \text{para } i = 0, \dots, m-1$$

- Dada uma chave  $k$ , primeiro sondamos  $T[h'(k)]$ , depois  $T[h'(k)+1]$ , e assim por diante até  $T[m-1]$ ;



Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando função  $h'(k)=k \bmod 11$ .

INSERÇÃO:

0	1	2	3	4	5	6	7	8	9	10

TENTATIVAS:



Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando função  $h'(k)=k \bmod 11$ .

INSERÇÃO: 10

0	1	2	3	4	5	6	7	8	9	10

TENTATIVAS: 0

$$\begin{aligned}h(10,0) &= (h'(10) + 0) \bmod 11 \\&= 10 \bmod 11 \\&= 10\end{aligned}$$

Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando função  $h'(k)=k \bmod 11$ .

INSERÇÃO: 10

0	1	2	3	4	5	6	7	8	9	10
										10

TENTATIVAS: 0

$$\begin{aligned}h(10,0) &= (h'(10) + 0) \bmod 11 \\&= 10 \bmod 11 \\&= 10\end{aligned}$$

Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h'(k)=k \bmod 11$ .

INSERÇÃO: 22

0	1	2	3	4	5	6	7	8	9	10
										10

TENTATIVAS: 0

$$\begin{aligned}h(22,0) &= (h'(22) + 0) \bmod 11 \\&= 0 \bmod 11 \\&= 0\end{aligned}$$



Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h(k)=k \bmod 11$ .

INSERÇÃO: 22

0	1	2	3	4	5	6	7	8	9	10
										10

TENTATIVAS: 0

$$\begin{aligned}h(22,0) &= (h'(22) + 0) \bmod 11 \\&= 0 \bmod 11 \\&= 0\end{aligned}$$

Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h'(k)=k \bmod 11$ .

INSERÇÃO: 22

0	1	2	3	4	5	6	7	8	9	10
22										10

TENTATIVAS: 0

$$\begin{aligned}h(22,0) &= (h'(22) + 0) \bmod 11 \\&= 0 \bmod 11 \\&= 0\end{aligned}$$

Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h'(k)=k \bmod 11$ .

INSERÇÃO: 31

0	1	2	3	4	5	6	7	8	9	10
22										10

TENTATIVAS: 0

$$\begin{aligned}h(31,0) &= (h'(31) + 0) \bmod 11 \\&= 9 \bmod 11 \\&= 9\end{aligned}$$



Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h(k)=k \bmod 11$ .

INSERÇÃO: 31

0	1	2	3	4	5	6	7	8	9	10
22										10

TENTATIVAS: 0

$$\begin{aligned}h(31,0) &= (h'(31) + 0) \bmod 11 \\&= 9 \bmod 11 \\&= 9\end{aligned}$$

Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h(k)=k \bmod 11$ .

INSERÇÃO: 31

0	1	2	3	4	5	6	7	8	9	10
22									31	10

TENTATIVAS: 0

$$\begin{aligned}h(31,0) &= (h'(31) + 0) \bmod 11 \\&= 9 \bmod 11 \\&= 9\end{aligned}$$



Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h(k)=k \bmod 11$ .

INSERÇÃO: 4

0	1	2	3	4	5	6	7	8	9	10
22									31	10

TENTATIVAS: 0

$$\begin{aligned}h(4,0) &= (h'(4) + 0) \bmod 11 \\&= 4 \bmod 11 \\&= 4\end{aligned}$$



Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h(k)=k \bmod 11$ .

INSERÇÃO: 4

0	1	2	3	4	5	6	7	8	9	10
22									31	10

TENTATIVAS: 0

$$\begin{aligned}h(4,0) &= (h'(4) + 0) \bmod 11 \\&= 4 \bmod 11 \\&= 4\end{aligned}$$

Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h'(k)=k \bmod 11$ .

INSERÇÃO: 4

0	1	2	3	4	5	6	7	8	9	10
22				4					31	10

TENTATIVAS: 0

$$\begin{aligned}h(4,0) &= (h'(4) + 0) \bmod 11 \\&= 4 \bmod 11 \\&= 4\end{aligned}$$

Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h'(k)=k \bmod 11$ .

INSERÇÃO: 15

0	1	2	3	4	5	6	7	8	9	10
22				4					31	10

TENTATIVAS: 0

$$\begin{aligned}h(15,0) &= (h'(15) + 0) \bmod 11 \\&= 4 \bmod 11 \\&= 4\end{aligned}$$



Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h'(k)=k \bmod 11$ .

INSERÇÃO: 15

Colisão

0	1	2	3	4	5	6	7	8	9	10
22				4					31	10

TENTATIVAS: 0

$$\begin{aligned}h(15,0) &= (h'(15) + 0) \bmod 11 \\&= 4 \bmod 11 \\&= 4\end{aligned}$$

Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h(k)=k \bmod 11$ .

**INSERÇÃO: 15****Colisão**

0	1	2	3	4	5	6	7	8	9	10
22				4					31	10

**TENTATIVAS: 1**

$$h(15,1) = (h'(15) + 1) \bmod 11$$



Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h'(k)=k \bmod 11$ .

INSERÇÃO: 15

Colisão

0	1	2	3	4	5	6	7	8	9	10
22				4					31	10

TENTATIVAS: 1



$$\begin{aligned}h(15,1) &= (h'(15) + 1) \bmod 11 \\&= (4 + 1) \bmod 11 \\&= 5 \bmod 11 = 5\end{aligned}$$

Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h'(k)=k \bmod 11$ .

INSERÇÃO: 15

LIVRE

0	1	2	3	4	5	6	7	8	9	10
22				4					31	10

TENTATIVAS: 1



$$\begin{aligned} h(15,1) &= (h'(15) + 1) \bmod 11 \\ &= (4 + 1) \bmod 11 \\ &= 5 \bmod 11 = 5 \end{aligned}$$



Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h'(k)=k \bmod 11$ .

INSERÇÃO: 15

LIVRE

0	1	2	3	4	5	6	7	8	9	10
22				4	15				31	10

TENTATIVAS: 1



$$\begin{aligned} h(15,0) &= (h'(15) + 1) \bmod 11 \\ &= (4 + 1) \bmod 11 \\ &= 5 \bmod 11 = 5 \end{aligned}$$

Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h'(k)=k \bmod 11$ .

INSERÇÃO: 28

0	1	2	3	4	5	6	7	8	9	10
22				4	15				31	10

TENTATIVAS: 0

$$\begin{aligned}h(28,0) &= (h'(28) + 0) \bmod 11 \\&= (6 + 0) \bmod 11 \\&= 6 \bmod 11 = 6\end{aligned}$$



Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h'(k)=k \bmod 11$ .

INSERÇÃO: 28

0	1	2	3	4	5	6	7	8	9	10
22				4	15				31	10

TENTATIVAS: 0

$$\begin{aligned}h(28,0) &= (h'28) + 0) \bmod 11 \\&= (6 + 0) \bmod 11 \\&= 6 \bmod 11 = 6\end{aligned}$$

Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h(k)=k \bmod 11$ .

INSERÇÃO: 28

0	1	2	3	4	5	6	7	8	9	10
22				4	15	28			31	10

TENTATIVAS: 0

$$\begin{aligned} h(28,0) &= (h(28) + 0) \bmod 11 \\ &= (6 + 0) \bmod 11 \\ &= 6 \bmod 11 = 6 \end{aligned}$$

Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h'(k)=k \bmod 11$ .

INSERÇÃO: 59

0	1	2	3	4	5	6	7	8	9	10
22				4	15	28			31	10

TENTATIVAS: 0

$$\begin{aligned}h(59,0) &= (h'(59) + 0) \bmod 11 \\&= (4 + 0) \bmod 11 \\&= 4 \bmod 11 = 4\end{aligned}$$

Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h(k)=k \bmod 11$ .

INSERÇÃO: 59

Colisão

0	1	2	3	4	5	6	7	8	9	10
22				4	15	28			31	10

TENTATIVAS: 0

$$\begin{aligned}h(59,0) &= (h'(59) + 0) \bmod 11 \\&= (4 + 0) \bmod 11 \\&= 4 \bmod 11 = 4\end{aligned}$$

Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h(k)=k \bmod 11$ .

INSERÇÃO: 59

Colisão

0	1	2	3	4	5	6	7	8	9	10
22				4	15	28			31	10

TENTATIVAS: 1

$$\begin{aligned}h(59,1) &= (h'(59) + 1) \bmod 11 \\&= (4 + 1) \bmod 11 \\&= 5 \bmod 11 = 5\end{aligned}$$

Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h(k)=k \bmod 11$ .

INSERÇÃO: 59

Colisão

0	1	2	3	4	5	6	7	8	9	10
22				4	15	28			31	10

TENTATIVAS: 1



$$\begin{aligned} h(59,0) &= (h'(59) + 1) \bmod 11 \\ &= (4 + 1) \bmod 11 \\ &= 5 \bmod 11 = 5 \end{aligned}$$



Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h(k)=k \bmod 11$ .

INSERÇÃO: 59

Colisão

0	1	2	3	4	5	6	7	8	9	10
22				4	15	28			31	10

TENTATIVAS: 2



$$\begin{aligned} h(59,2) &= (h(59) + 2) \bmod 11 \\ &= (4 + 2) \bmod 11 \\ &= 6 \bmod 11 = 6 \end{aligned}$$

Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h(k)=k \bmod 11$ .

INSERÇÃO: 59

0	1	2	3	4	5	6	7	8	9	10
22				4	15	28			31	10

Colisão

TENTATIVAS: 2



$$\begin{aligned} h(59,2) &= (h(59) + 2) \bmod 11 \\ &= (4 + 2) \bmod 11 \\ &= 6 \bmod 11 = 6 \end{aligned}$$

Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h(k)=k \bmod 11$ .

INSERÇÃO: 59

Colisão

0	1	2	3	4	5	6	7	8	9	10
22				4	15	28			31	10

TENTATIVAS: 3



$$\begin{aligned} h(59,3) &= (h(59) + 3) \bmod 11 \\ &= (4 + 3) \bmod 11 \\ &= 7 \bmod 11 = 7 \end{aligned}$$

Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h(k)=k \bmod 11$ .

INSERÇÃO: 59

0	1	2	3	4	5	6	7	8	9	10
22				4	15	28			31	10

LIVRE

TENTATIVAS: 3



$$\begin{aligned}h(59,3) &= (h(59) + 3) \bmod 11 \\&= (4 + 3) \bmod 11 \\&= 7 \bmod 11 = 7\end{aligned}$$

Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h(k)=k \bmod 11$ .

INSERÇÃO: 59

0	1	2	3	4	5	6	7	8	9	10
22				4	15	28	59		31	10

LIVRE

TENTATIVAS: 3



$$\begin{aligned} h(59,0) &= (h(59) + 3) \bmod 11 \\ &= (4 + 3) \bmod 11 \\ &= 7 \bmod 11 = 7 \end{aligned}$$

Endereçamento Aberto:

## Sondagem Linear

- Exemplo: Insira as chaves {10,22,31,4,15,28,59} em uma tabela T de tamanho 11 com sondagem linear usando a função  $h'(k)=k \bmod 11$ .

FINAL

0	1	2	3	4	5	6	7	8	9	10
22				4	15	28	59		31	10

**Desvantagem:** É suscetível ao agrupamento primário, isto é, são construídas longas sequências de posições ocupadas, o que degrada o desempenho da busca.

## Sondagem Quadrática

- A sondagem quadrática utiliza uma função hash da forma:

$$h(k,i) = (h'(k) + c_1 i + c_2 i^2) \bmod m$$

onde,  $h'(k)$  é uma função de hash auxiliar,  $c_1$  e  $c_2$  são constantes não nulas e  $i=0,1,2,\dots,m-1$ .

- A posição inicial sondada é  $T[h'(k)]$ ; posições posteriores são deslocadas por quantidades que dependem de forma quadrática do número da sondagem  $i$ .
- Funciona melhor que a sondagem linear, mas para fazer pleno uso da tabela hash, os valores de  $c_1$ ,  $c_2$  e  $m$  são restritos.

## Sondagem Dupla

- Um dos melhores métodos disponíveis, pois as permutações produzidas têm muitas características de permutações escolhidas aleatoriamente.
- O hash duplo usa uma função hash da forma:

$$h(k,i) = (h_1(k) + ih_2(k)) \bmod m,$$

onde  $h_1$  e  $h_2$  são funções hash auxiliares e  $i = 0, 1, 2, \dots, m-1$ ;

- Sondagem inicial vai a posição  $T[h_1(k)]$ ; posições sucessivas são deslocadas em relação às posições anteriores pela quantidade  $h_2(k)$ , módulo  $m$ .



# Sondagem Dupla

## Questão importante: como escolher $h_1$ , $h_2$ e $m$ ?

- O valor de  $h_2(k)$  e o tamanho de  $m$  da tabela hash devem ser primos entre si.
- Como fazer isso?
  - *Fazer  $m$  ser potência de dois e projetar  $h_2$  para sempre retornar um número ímpar.*
  - *Fazer  $m$  primo e projetar  $h_2$  para sempre retornar um número inteiro positivo menor que  $m$ .*

### Exemplo para o segundo caso

Escolher  $m$  primo e fazer:

$$h_1(k) = k \bmod m,$$

$$h_2(k) = 1 + (k \bmod m'),$$

onde o valor de  $m'$  escolhido é ligeiramente menor que  $m$ .

→ Para  $k = 123456$ ,  $m = 701$  e  $m' = 700$ , tem-se:

◆  $h_1(123456) = 80$

◆  $h_2(123456) = 257$

→ A primeira posição sondada é de número 80; as demais estão separadas por 257 posições.

# Sondagem Dupla

**Exemplo:** Inserir as chaves: 74,92,32,70,47

(1) Escolher  $m = 7$  e  $m' = 5$

(2) Função hash:

$$h(k,i) = ( h_1(k) + i h_2(k) ) \bmod 7$$

$$h_1(k) = k \bmod 7$$

$$h_2(k) = 1 + k \bmod 5$$

T	
	0
	1
	2
	3
	4
	5
	6



# Sondagem Dupla

**Exemplo:** Inserir as chaves:

(1) Escolher  $m = 7$  e  $m' = 5$

(2) Função hash:  $h(k,i) = (h_1(k) + i h_2(k)) \bmod 7$   
 $h_1(k) = k \bmod 7$   
 $h_2(k) = 1 + k \bmod 5$

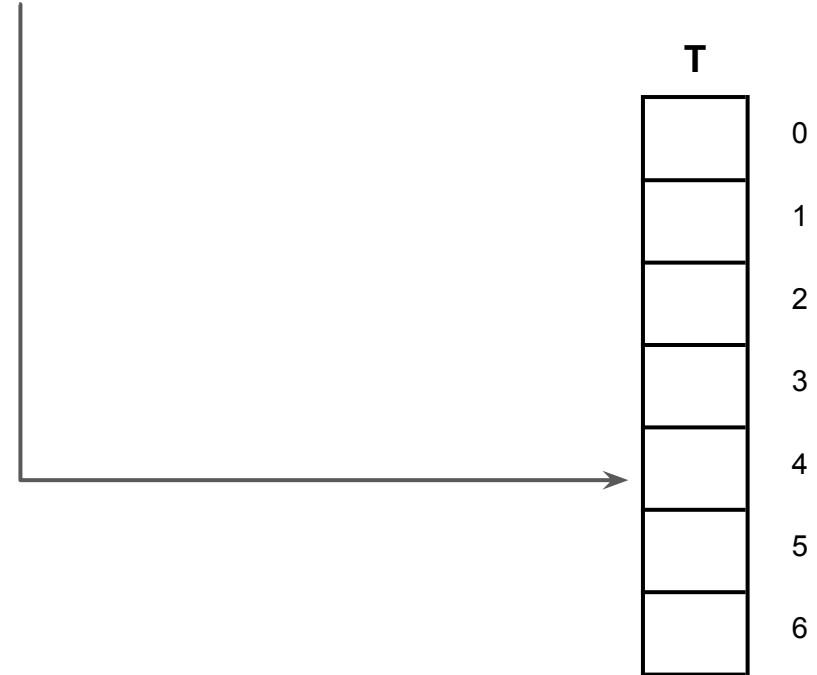
**Inserção: 74**

$$h_1(74) = 74 \bmod 7 = 4$$

$$h_2(74) = 1 + 74 \bmod 5 = 1 + 4 = 5$$

$$h(74,0) = (4 + 0 \cdot 5) \bmod 7 = 4$$

74, 92, 32, 70, 47



# Sondagem Dupla

**Exemplo:** Inserir as chaves:

(1) Escolher  $m = 7$  e  $m' = 5$

(2) Função hash:  $h(k,i) = (h_1(k) + i h_2(k)) \bmod 7$   
 $h_1(k) = k \bmod 7$   
 $h_2(k) = 1 + k \bmod 5$

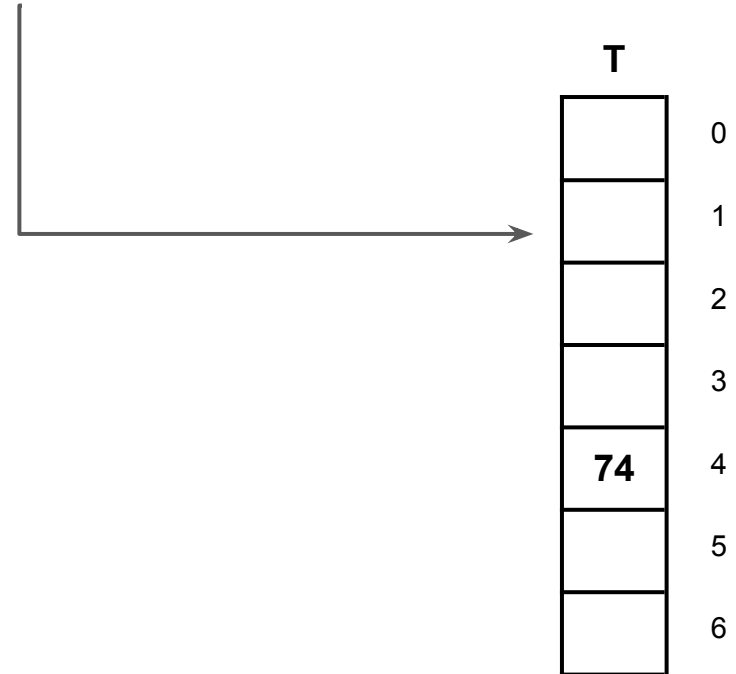
## Inserção: 92

$$h_1(92) = 92 \bmod 7 = 1$$

$$h_2(92) = 1 + 92 \bmod 5 = 1 + 2 = 3$$

$$h(92,0) = (1 + 0 \cdot 3) \bmod 7 = 1$$

74, **92**, 32, 70, 47



# Sondagem Dupla

**Exemplo:** Inserir as chaves:

(1) Escolher  $m = 7$  e  $m' = 5$

(2) Função hash:  $h(k,i) = (h_1(k) + i h_2(k)) \bmod 7$   
 $h_1(k) = k \bmod 7$   
 $h_2(k) = 1 + k \bmod 5$

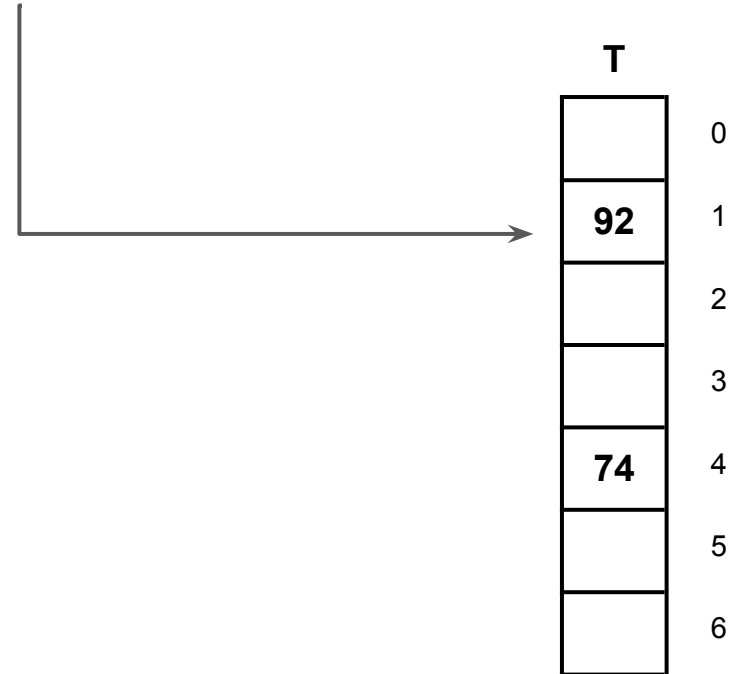
**Inserção: 92**

$$h_1(92) = 92 \bmod 7 = 1$$

$$h_2(92) = 1 + 92 \bmod 5 = 1 + 2 = 3$$

$$h(92,0) = (1 + 0 \cdot 3) \bmod 7 = 1$$

74, **92**, 32, 70, 47



# Sondagem Dupla

**Exemplo:** Inserir as chaves:

(1) Escolher  $m = 7$  e  $m' = 5$

(2) Função hash:  $h(k,i) = (h_1(k) + i h_2(k)) \bmod 7$   
 $h_1(k) = k \bmod 7$   
 $h_2(k) = 1 + k \bmod 5$

**Inserção: 32**

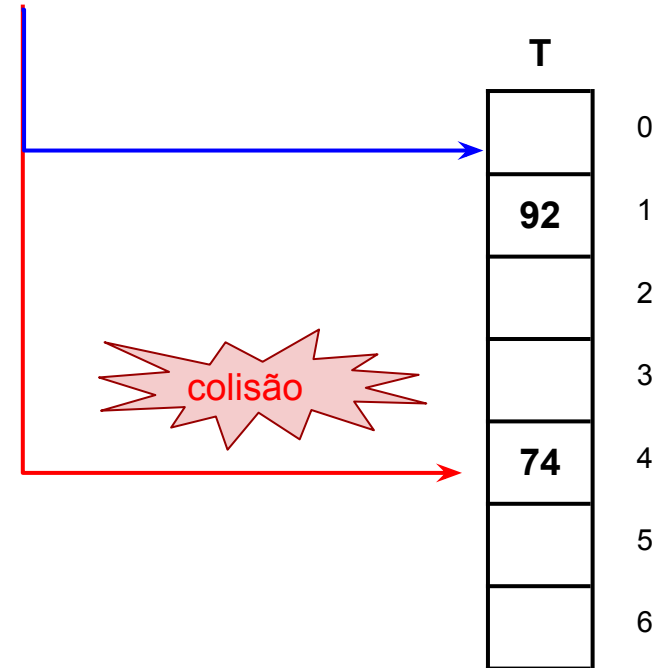
$$h_1(32) = 32 \bmod 7 = 4$$

$$h_2(32) = 1 + 32 \bmod 5 = 1 + 2 = 3$$

$$h(32,0) = (4 + 0 \cdot 3) \bmod 7 = 4$$

$$h(32,1) = (4 + 1 \cdot 3) \bmod 7 = 7 \bmod 7 = 0$$

74, 92, **32**, 70, 47



# Sondagem Dupla

**Exemplo:** Inserir as chaves:

(1) Escolher  $m = 7$  e  $m' = 5$

(2) Função hash:  $h(k,i) = (h_1(k) + i h_2(k)) \bmod 7$   
 $h_1(k) = k \bmod 7$   
 $h_2(k) = 1 + k \bmod 5$

**Inserção: 70**

$$h_1(70) = 70 \bmod 7 = 0$$

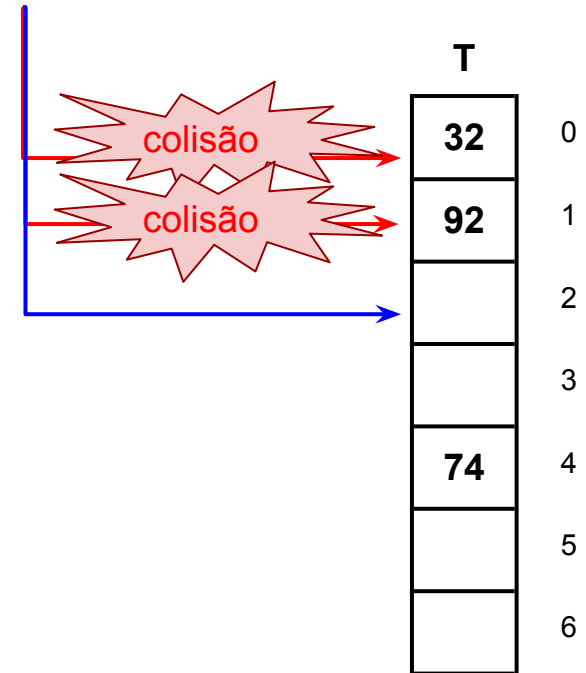
$$h_2(70) = 1 + 70 \bmod 5 = 1 + 0 = 1$$

$$h(70,0) = (0 + 0 \cdot 1) \bmod 7 = 0$$

$$h(70,1) = (0 + 1 \cdot 1) \bmod 7 = 1 \bmod 7 = 1$$

$$h(70,2) = (0 + 2 \cdot 1) \bmod 7 = 2 \bmod 7 = 2$$

74, 92, 32, **70**, 47



# Sondagem Dupla

**Exemplo:** Inserir as chaves:

74, 92, 32, 70, **47**

(1) Escolher  $m = 7$  e  $m' = 5$

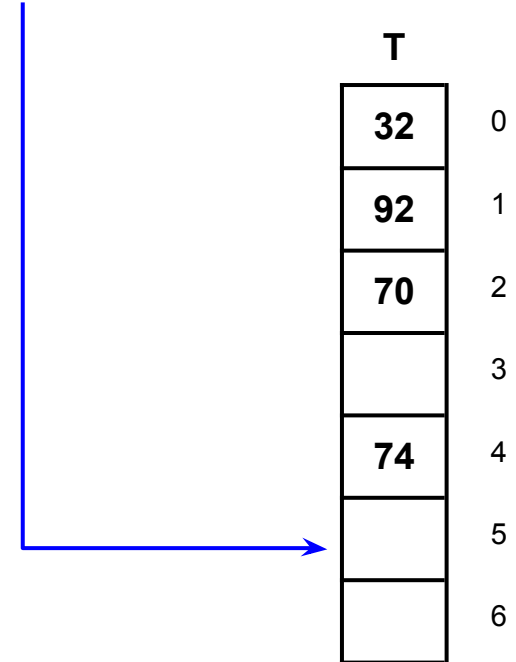
(2) Função hash:  $h(k,i) = (h_1(k) + i h_2(k)) \bmod 7$   
 $h_1(k) = k \bmod 7$   
 $h_2(k) = 1 + k \bmod 5$

**Inserção: 47**

$$h_1(47) = 47 \bmod 7 = 5$$

$$h_2(47) = 1 + 47 \bmod 5 = 1 + 2 = 3$$

$$h(47,0) = (5 + 0 \cdot 3) \bmod 7 = 5$$





# Sondagem Dupla

**Exemplo:** Inserir as chaves:

74, 92, 32, 70, **47**

(1) Escolher  $m = 7$  e  $m' = 5$

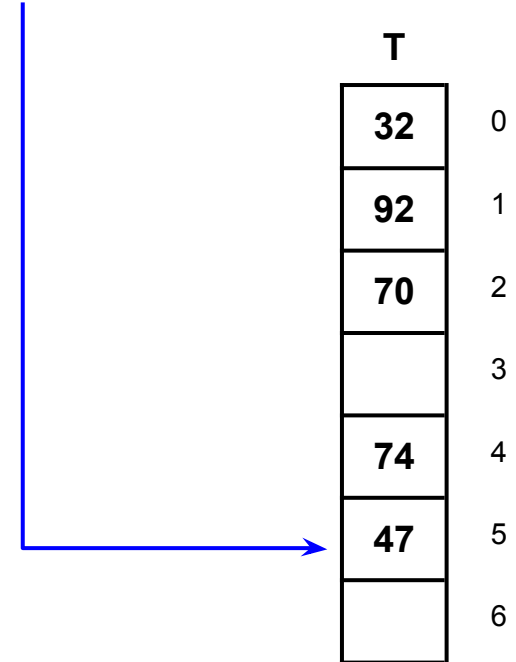
(2) Função hash:  $h(k,i) = (h_1(k) + i h_2(k)) \bmod 7$   
 $h_1(k) = k \bmod 7$   
 $h_2(k) = 1 + k \bmod 5$

**Inserção: 47**

$$h_1(47) = 47 \bmod 7 = 5$$

$$h_2(47) = 1 + 47 \bmod 5 = 1 + 2 = 3$$

$$h(47,0) = (5 + 0 \cdot 3) \bmod 7 = 5$$



## Remoção em Endereçamento Aberto

- Suponha que queremos remover o elemento de chave  $k$  da tabela.
- A partir da posição  $h(k)$  temos que procurar  $k$  usando a sequência de sondagem para descobrir a posição verdadeira deste. Suponha que ele esteja na posição  $j$ .
- Não podemos simplesmente colocar  $T[j] = NULL$ . **Por quê?**
  - **Influência na busca dos elementos na hash!**

# Remoção usando Sondagem Dupla

**Exemplo:** **Remover a chave 74** e pesquisar a chave 32.

(1) Escolher  $m = 7$  e  $m' = 5$

(2) Função hash:  $h(k,i) = (h_1(k) + i h_2(k)) \bmod 7$   
 $h_1(k) = k \bmod 7$   
 $h_2(k) = 1 + k \bmod 5$

**Remover: 74**

$$h_1(74) = 74 \bmod 7 = 4$$

$$h_2(74) = 1 + 74 \bmod 5 = 1 + 4 = 5$$

$$h(74,0) = (4 + 0 \cdot 5) \bmod 7 = 4$$

T	
32	0
92	1
70	2
	3
	4
47	5
	6



# Remoção usando Sondagem Dupla

**Exemplo:** Remover a chave 74 e **pesquisar a chave 32**.

(1) Escolher  $m = 7$  e  $m' = 5$

(2) Função hash:  $h(k,i) = (h_1(k) + i h_2(k)) \bmod 7$   
 $h_1(k) = k \bmod 7$   
 $h_2(k) = 1 + k \bmod 5$

**Pesquisar chave: 32**

$$h_1(32) = 32 \bmod 7 = 4$$

$$h_2(32) = 1 + 32 \bmod 5 = 1 + 2 = 3$$

$$h(32,0) = (4 + 0 \cdot 3) \bmod 7 = 4$$

Posição vazia!  
A busca retornaria a  
chave não encontrada!

T	
32	0
92	1
70	2
	3
	4
47	5
	6

## Remoção em Endereçamento Aberto

- Suponha que queremos remover o elemento de chave  $k$  da tabela.
- A partir da posição  $h(k)$  temos que procurar  $k$  usando a sequência de sondagem para descobrir a posição verdadeira deste. Suponha que ele esteja na posição  $j$ . Não podemos simplesmente colocar  $T[j] = \text{NULL}$ . **Por que?**
  - Influência na busca dos elementos na hash!
  - **Uma chave não pode ser removida de fato pois perderia a sequência de sondagens**
- Podemos resolver isso colocando um marcador REMOVIDO em vez de NULL. Isto indica que uma busca não deve parar neste ponto. Além disso, para a inserção, esta corresponderia a uma posição livre.

## Tabela Hash - Discussão

A técnica de hashing é mais utilizada nos casos em que existem muito mais buscas do que inserções de registros

# Tabela hash

## ➤ Referências:

- ZIVIANI, NIVIO. Projeto de Algoritmos com Implementação em Java e C++. Cengage Learning. 2007.
- LEISERSON, C. E.; STEIN, C.; RIVEST, R. L., CORMEN, T.H. Algoritmos: Teoria e Prática. Tradução da 2a. edição americana. Editora Campus, 2002.
- PREISS, Bruno. Estrutura de Dados e Algoritmos. Elsevier. 2001.

## ➤ Extras

- Hash para arquivos extensíveis
- Leitura do seção 10.6.1 do Livro
  - Drozdek, A. Estrutura de Dados e Algoritmos em C++ – Tradução da 4ª edição norte-americana: Cengage Learning Brasil, 2018. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788522126651/>. Acesso em: 11 Sep 2020

## Exercício 1:

Uma tabela hash de tamanho 10 usa endereçamento aberto com função hash  $h(k) = k \bmod 10$  e sondagem linear. Após 6 inserções em uma tabela hash vazia, tem-se a seguinte configuração

0	1	2	3	4	5	6	7	8	9
		42	23	34	52	46	33		

Qual das opções a seguir fornece uma ordem possível em que as chaves poderiam ter sido inseridas na tabela?

- (a) 46, 42, 34, 52, 23, 33
- (b) 34, 42, 23, 52, 33, 46
- (c) 46, 34, 42, 23, 52, 33
- (d) 42, 46, 33, 23, 34, 52



## Exercício 2:

Considere a inserção das chaves 10, 22, 31, 4, 15, 28, 17, 88, 59 em uma tabela hash de comprimento  $M = 11$  usando o endereçamento aberto com sondagem dupla com

- $h_1(k) = k \bmod m$
- $h_2(k) = 1 + (k \bmod (m - 1))$

Quantas colisões ocorreram nas inserções acima?